

Jazyk AdAs

SEMESTRÁLNÍ PRÁCE Z PŘEDMĚTU KIV/FJP

MARTIN BROŽEK, JAROSLAV HRUBÝ

Datové typy

`int` – celé číslo

`bool` – pravdivostní hodnota

`int[<velikost>]`, `bool[<velikost>]` – pole hodnot

Práce s datovými typy

Deklarace

- `<datový_typ> <identifikátor>;`

Přiřazení

- `<datový_typ> <identifikátor> = <hodnota>;`
- `<identifikátor> = <hodnota>;`

Konstanta

- `const <datový_typ> <identifikátor> = <hodnota>;`

Přístup do pole

- `<identifikátor>[<index>]`

Práce s datovými typy

Aritmetické operace

- $(+, a, b)$
- $(-, a, b)$
- $(*, a, b)$
- $(/, a, b)$

Logické operace

- (AND, a, b)
- (OR, a, b)
- $(!, (<=, a, b))$

Jazykové konstrukce

Cykly:

- `while (<podmínka>) { <tělo> }`
- `do { <tělo> } while (<podmínka>)`
- `for <identifikátor> in <identifikátor_pole> { <tělo> }`
- `for <identifikátor> in <horní_hranice|identifikátor> { <tělo> }`

Podmínky:

- `if (<podmínka>) { <tělo> } else { <tělo> }`
- `switch <identifikátor> { case <hodnota>: { <tělo> }
default: { <tělo> } }`

Ternární operátor:

- `(<, a, b) ? a : b;`

Jazykové konstrukce

Definice podprogramů:

```
function <návratová_hodnota> <název_podprogramu>(<parametry>) { <tělo> }
```

Volání podprogramů:

```
(<název_podprogramu>, <parametry>)
```

Průběh překladu

- 1) Kontrola vstupních parametrů
- 2) Průchod syntaktického stromu
- 3) Naplnění připravených objektů
- 4) Shromažďování instrukcí
- 5) Zápis instrukcí do souboru

Děkujeme za pozornost
